Title: Summer 2021 Internship Report

Author(s): Jayawardana, Pathiranage Yasith Milinda

Intended for: Report

Issued: 2021-08-24

Disclaimer:
Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by Triad National Security, LLC for the National Nuclear Security Administration of U.S. Department of Energy under contract 89233218CNA000001. By approving this article, the publisher recognizes that the U.S. Government retains nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Summer 2021 Internship Report

Name:	Yasith Jayawardana
UIN:	01130624
Email:	yasith@cs.odu.edu
Course:	CS667 Cooperative Education
Date:	August 17, 2021

I have reviewed this report and ensured that it is an accurate description of the duties and functions of the assigned position during the work period. I confirm that the information discussed herein does not violate company regulations or confidences and is appropriate for release.

(Add LA-UR # here)

…………………………………………
Dr. Martin Klein
Research and Prototyping Team
LANL Research Library

Section I: Position Information

- **Title:** Summer Research Intern
- **Organization:** Los Alamos National Laboratory, Los Alamos, NM
- **Department:** Research and Prototyping Team – Research Library
- **Supervisor:** Dr. Martin Klein

About the Organization

Los Alamos National Laboratory (LANL) is a Federally Funded Research and Development Center (FFRDC) with a mission to solve national security challenges through cutting-edge scientific research. Its strategic plan reflects U.S. priorities spanning nuclear security, intelligence, defense, emergency response, nonproliferation, counterterrorism, energy security, emerging threats, and environmental management. Every year, LANL onboards approximately 2000 students to work on various scientific applications. These students are given the flexibility to work on different projects and gain hands-on experience in solving various scientific research problems.

The Research and Prototyping Team (Proto Team) of the Research Library of LANL explores various aspects of scholarly communication and open science in the digital age. The team focuses on information infrastructure, information interoperability, and long-term persistence of scholarly records. Recently, the team led the development and standardization of the Memento interoperability framework to access past versions of the Web resources.

About the Project

During this internship program, I was assigned with developing a Web application to *robustify* web resources referenced by links (URI-Rs) in PDFs. It was intended for use in LANL Research

Library systems such as RASSTI (Review and Approval System for Scientific and Technical Information), where users submit scholarly PDFs that may contain URI-Rs. For such PDFs, the application should, using Web Archives, create robust snapshots (i.e., Mementos) of the web resources referenced by URI-Rs, and notify this robustification to LANL Research Library systems. The Mementos were to be created using the Robust Links Service, and the notifications were to be sent using the Linked Data Notifications (LDN) Protocol.

Section II: Duties/Responsibilities

I was fully responsible for carrying out the project, which involved prototyping the URI-R extraction with multiple tools and technologies, evaluating their performance and speed, and using those results to formulate a viable plan of action to develop the Web application.

Onboarding and Meetings

The program started with a week-long orientation, where I attended training sessions to familiarize with LANL and its processes. In the second week, I began working on my assigned project. Throughout this program, I attended weekly 1-1 meetings with my supervisor (Dr. Martin Klein) and bi-weekly meetings with the entire team. The 1-1 meetings were to report my progress, obtain feedback, resolve issues, and plan for the next week. The team meetings were for team-wide announcements and for all team members to present what they are working on. Throughout the project, I corresponded daily with Dr. Klein via Email and Hangouts.

Research and Evaluation

With the guidance of Dr. Klein, I first familiarized with the concepts of Web Archiving, LDN, and Regular Expressions. Next, I explored tools and techniques to extract URI-Rs from PDFs. I first implemented an annotation-based URI-R extractor (using PyPDF2), and four text-based URI-R extractors using GROBID, PDFMiner and Regular Expressions. I also curated a dataset with 8 sample PDFs and their URI-Rs to perform quantitative evaluation. I observed that the text-based extractors picked some URI-Rs that the annotation-based extractor missed, but also picked partial/invalid URI-Rs as well. After discussing this issue with Dr. Klein, I implemented four two-step URI-R extractors by combining each text-based extractor with the annotation-based extractor. In the meantime, I built another extractor based on the PDF viewer of Google Chrome (i.e., PDFIUM). Upon evaluation, I found that the combined PyPDF2+PDFIUM extractor yielded the best performance, with a considerable speed improvement over the other extractors. Based on these findings, I selected this URI-R extractor to implement the web application.

Building the Web Application

Next, I progressed onto building the web application using the PyPDF2+PDFIUM extractor. I first created an API using the Flask framework. I first implemented API functions to 1) upload a PDF, 2) extract URI-Rs from a PDF, 3) robustify URI-Rs (through the Robust Links Service) and get their URI-Ms (i.e., the URIs of their Mementos), and 4) send a LDN with URI-R → URI-M mappings to a LDN Inbox. Next, I developed Web pages using HTML and Javascript to use this API on a Web Browser. Based on Dr. Klein's feedback, I made iterative improvements to this Web Application by fixing issues and refining the user experience. This solution can now be integrated into LANL Research Library systems that support LDN. Upon receiving an LDN, they could, for instance, display an HTML overview of the PDF, its URI-Rs, and their *robust* URI-Ms.

Section III: Progression

Prior to this internship program, I only had limited knowledge on Web Archiving, LDN, and Regular Expressions. Over time, I learned the basics of these concepts, and how to apply them to build the expected solution. Before developing the solution, I performed a comprehensive exploration of the tools and technologies related to the project and evaluated them from different perspectives in context of the problem. Through this, I got a holistic understanding of the problem and its challenges, which in turn, helped me to tackle it objectively with evaluation results as proof.

Section IV: Academic Relevance

Coursework

In my project, I first had to explore available tools/technologies and evaluate them objectively to find a strategy that works the best. Based on my research, I discovered that URI-Rs in PDFs may exist as annotations, and that, using PyPDF2, I could easily extract such URI-Rs without the need to harvest URI-Rs from PDF text. Based on this finding, I first implemented an annotation-based URI-R extractor using PyPDF2. Next, I curated a dataset with 8 sample PDFs and their URI-Rs to objectively evaluate any extractor. Upon evaluating the PyPDF2 extractor, I noticed that it missed all non-annotated URI-Rs. This led me to build four text-based URI-R extractors using GROBID, PDFMiner and Regular Expressions. I paired GROBID and PDFMiner with two Regular Expressions that I found online, and thus create four text-based extractors. In the meantime, I was also studying how PDF viewers (such as Adobe Acrobat DC, macOS Preview) and Web browsers (such as Chrome and Safari) identify URI-Rs in PDFs. Based on my findings, I built another text-based URI-R extractor using Google Chrome's PDF viewer, PDFIUM. I observed that the text-based extractors picked some URI-Rs that the annotation-based extractor missed, but also picked partial/invalid URI-Rs (e.g, the first line of a newline-seperated URI-R). After discussing this issue with Dr. Klein, we decided to implement two-step URI-R extractors that perform both annotation-based and text-based URI-R extraction. Here, I picked all URI-Rs extracted from PDF annotations, but only picked the URI-Rs extracted from PDF text that did not (even partially) match the URI-Rs that were already picked. Upon evaluation, I found that the combined PyPDF2+PDFIUM extractor performed the best overall. The PyPDF2+PDFMiner and PyPDF2+GROBID extractors performed slightly better on a few PDFs. However, the PyPDF2+PDFIUM extractor ran significantly faster than the two-step extractors. These findings gave conclusive proof in favor of the PyPDF2+PDFIUM extractor. Overall, I performed an entire study where I researched, developed, evaluated, concluded, and documented the best possible solution to the problem being addressed.

Professional Literature

Due to the dynamic nature of the web, links to web resources could break over time (i.e., *link rot*). In the scholarly world, publications may cite web resources, thereby rendering them susceptible to link rot. One study [1] analyzed the web resources cited in 3,024,986 publications across different disciplines, time-periods, and types. They found that 22.7% of these publications had cited at least one web resource, totaling 3,656,553 links. Thus, web resources are widely cited in publications, and thereby, should be an important consideration for digital preservation. They also identified two challenges in extracting links from PDF documents, namely: 1) line breaks appearing within URLs, and 2) certain characters (e.g., "_") appearing as in-text images. In my internship project, I tried to address these challenges by augmenting the extracted PDF text with line breaks removed. Doing so resulted in a better recall, at the cost of reduced precision.

Another study [2] found that, out of 193,955 links referenced by publications in the Elsevier scholarly article collection, 36.2% of links were *rotten*, but only 62.3% of *rotten* links were archived; the remaining 37.7% of *rotten* links are permanently inaccessible for future researchers. As a solution, they proposed a framework to predict whether a link would rot and pro-actively archive them. Through simulation, they were able to archive 84.8% of *rotten* links (compared to 62.3%). These findings reflect the motivation behind my internship project.

While the Web application that I developed during the internship program was targeted at PDFs submitted via LANL Research Library systems, it is applicable to any scholarly publication system; publishers can utilize this application to pro-actively archive web resources referenced in scholarly PDFs, and thereby retain a *robust* copy of them for future access. I plan to investigate this idea in my future research. Overall, I believe the practical experience that I gained, and the professional network that I built through this internship program is invaluable for my career development, and I hope to apply what I learned during this internship to improve my ongoing and future research.

Though the code that I developed is currently not publicly available, my supervisor and I plan to release an open-source version of it in the future.

Section V: Future Projections

Through this internship, I gained practical experience in three areas of Computer Science, namely: Text Mining, Web Archiving, and Systems Integration. On Text Mining, I learned how to extract URIs from PDFs using several PDF processing tools, and how to extract portions of text using Regular Expressions. On Web Archiving, I learned how to robustify links using the Robust Links Service, and the Memento framework in general. On Systems Integration, I learned how to build interoperable systems using Linked Data Notifications.

Prior to this internship, I had no work experience in the US outside of ODU. The experience that I gained through this internship unlocks many career opportunities for me, including jobs titled Data Scientist, Data Engineer, Systems Engineer, and Research Scientist. I believe having this experience on my resume would boost my chances of finding career opportunities in the future. I've also updated my Personal Website, LinkedIn, Resume, and CV to reflect what I learned through this internship. I also got the opportunity to expand my professional network by collaborating with researchers and CS professionals at LANL, which, in turn, may provide future opportunities for an internship, postdoc, or even a scientific career.

Section VI: Conclusion

Amidst the uncertainties from COVID-19, I got the opportunity to work as a Summer Research Intern at LANL from June 2021 to August 2021. This was my first internship as a PhD student in the US, and I cannot be grateful enough for this opportunity. Though I worked remotely, the entire team was supportive, accommodating, and welcoming. I believe the practical experience that I gained through this internship is invaluable for my career development, and I would recommend any student to apply for internship positions at LANL. I would like to thank my PhD advisor, Dr. Sampath Jayarathna, for encouraging me to apply for an internship at LANL. I'm grateful for my internship supervisor, Dr. Martin Klein, who recommended me for this position and guided me at every step. I am honored and thankful to have worked with him and the diverse, multi-disciplinary team at LANL as a Summer Research Intern.

Section VII: Beneficial Suggestions

URI-R Extraction: In the beginning of my internship project, the idea was to use Regular Expressions to extract URI-Rs from PDFs. I initially used GROBID and PDFMiner to perform URI-R extraction in this manner. However, I found that PyPDF2 can extract annotated URI-Rs from PDFs with no false positives. I also experimented with PDFIUM, the PDF renderer of the Chromium Web browser. Upon evaluating them on 8 sample PDFs, I found that the combined

PyPDF2 and PDFIUM URI-R extractor performed the best. Based on this, I suggested to use this method instead of relying on Regular Expressions to extract URI-Rs. This suggestion proved to be useful, as the final solution was fast, self-contained, and extracted URI-Rs better than the alternatives.

Documentation: Initially I used Google Docs to create the project documentation. However, this proved to be difficult, especially when navigating the code and results in 1-1 meetings. Therefore, I suggested hosting the project privately on GitHub, to which Dr. Klein agreed. I also suggested using the Sphinx documentation library to generate the project documentation from code comments and Markdowns. Using Sphinx made it easy for me to document the project functionality and code. The project documentation thus generated, was more comprehensible than what I would have otherwise written on Google Docs.

Section VIII: Quotes / Photos / Videos

Figure 1: Teleworking for LANL

“Amidst the uncertainties from COVID-19, I got the opportunity to work as a Summer Research Intern at LANL from June 2021 to August 2021. This was my first internship as a PhD student in the US, and I cannot be grateful enough for this opportunity. Though I was teleworking, the entire team was supportive, accommodating, and welcoming. I believe the practical experience that I gained, and the professional network that I built through this internship is invaluable for my future, and I would recommend any student to apply for internship positions at LANL.”

References

[1] K. Zhou, R. Tobin and C. Grover, "Extraction and analysis of referenced web links in large-scale scholarly articles," in *IEEE/ACM Joint Conference on Digital Libraries*, London, 2014.

[2] K. Zhou, C. Grover, M. Klein and R. Tobin, "No More 404s: Predicting Referenced Link Rot in Scholarly Articles for Pro-active Archiving," in *Proceedings of the 15th ACM/IEEE-CS Joint Conference on Digital Libraries*, Knoxville, 2015.